Лабораторная работа 2

1. Преобразуйте представление ось ω – угол θ в кватернион

1.1. $\omega = [-0.1457,\ 0.5976,\ -0.7884], \quad \theta = 3.5112$
1.2. $\omega = [0.4928,\ 0.5435,\ 0.6795], \quad \theta = 3.5366$
1.3. $\omega = [-0.1784,\ 0.2396,\ 0.9543], \quad \theta = 1.8534$
1.4. $\omega = [-0.5780,\ -0.7786,\ -0.2442], \quad \theta = 1.2844$
1.5. $\omega = [0.7362,\ 0.0666,\ 0.6734], \quad \theta = 4.0863$
1.6. $\omega = [-0.6893,\ 0.6863,\ 0.2319], \quad \theta = 5.0171$
1.7. $\omega = [-0.1380,\ -0.8528,\ -0.5037], \quad \theta = 6.1800$
1.8. $\omega = [0.0351,\ 0.5640,\ -0.8251], \quad \theta = 2.4076$
1.9. $\omega = [0.6360,\ 0.1757,\ 0.7515], \quad \theta = 2.9780$
1.10. $\omega = [0.2821,\ 0.0936,\ -0.9548], \quad \theta = 1.8078$
1.11. $\omega = [0.8807,\ 0.4069,\ -0.2426], \quad \theta = 1.6467$
1.12. $\omega = [-0.4320,\ 0.3838,\ 0.8162], \quad \theta = 1.8309$
1.13. $\omega = [0.6514,\ 0.6601,\ 0.3740], \quad \theta = 4.2487$

2. Преобразуйте кватернион $Q = [Q_s,\ Q_x,\ Q_y,\ Q_z]$ в матрицу поворота R согласно формуле:

$$R = \begin{bmatrix} 1 - 2Q_y^2 - 2Q_z^2 & 2Q_xQ_y - 2Q_zQ_s & 2Q_xQ_z + 2Q_yQ_s \\ 2Q_xQ_y + 2Q_zQ_s & 1 - 2Q_x^2 - 2Q_z^2 & 2Q_yQ_z - 2Q_xQ_s \\ 2Q_xQ_z - 2Q_yQ_s & 2Q_yQ_z + 2Q_xQ_s & 1 - 2Q_x^2 - 2Q_y^2 \end{bmatrix}$$

2.1. $Q = [-0.4161,\ 0.3523,\ -0.3074,\ 0.7800]$
2.2. $Q = [0.9010,\ -0.0131,\ -0.3935,\ 0.1818]$
2.3. $Q = [-0.6497,\ -0.3817,\ -0.4074,\ 0.5159]$
2.4. $Q = [0.8238,\ 0.0256,\ 0.1482,\ 0.5466]$
2.5. $Q = [0.4707,\ 0.6699,\ 0.5226,\ 0.2377]$
2.6. $Q = [0.8826,\ 0.3873,\ -0.1206,\ -0.2376]$
2.7. $Q = [0.6442,\ -0.5851,\ -0.3146,\ 0.3791]$
2.8. $Q = [-0.3169,\ 0.1932,\ -0.6358,\ 0.6768]$
2.9. $Q = [-0.7757,\ 0.5270,\ -0.2611,\ -0.2290]$
2.10. $Q = [-0.1433,\ -0.5519,\ -0.6894,\ -0.4467]$
2.11. $Q = [-0.8954,\ 0.2335,\ -0.2158,\ 0.3116]$
2.12. $Q = [0.4678,\ -0.6865,\ 0.2708,\ 0.4863]$
2.13. $Q = [0.8540,\ 0.1173,\ 0.2956,\ 0.4118]$

3. Преобразуйте матрицу поворота R в представление ось $\omega = [\omega_x,\ \omega_y,\ \omega_z]$ – угол θ согласно формулам:

$$\theta = \operatorname{acos}\frac{\operatorname{trace}(R) - 1}{2}, \qquad \omega = \frac{1}{2\sin\theta}\,[R_{32} - R_{23},\ R_{13} - R_{31},\ R_{21} - R_{12}].$$

Несмотря на то, что одной матрице поворота R соответствуют два эквивалентных решения $(\omega,\ \theta)$ и $(-\omega,\ -\theta)$, задание ограничено случаем $\theta \in [0,\ \pi]$. Однако необходимо учесть особые точки, при которых

$2\sin\theta = 0$. При $\theta = 0$ существует бесконечное число решений, и в этом случае $\omega = [\mathrm{NaN}, \mathrm{NaN}, \mathrm{NaN}]$. При $\theta = \pi$ существует два решения, вычислить которые можно по формуле:

$$R = \begin{bmatrix} \omega_x^2\,\nu_\theta + c_\theta & \omega_x\,\omega_y\,\nu_\theta - \omega_z\,s_\theta & \omega_x\,\omega_z\,\nu_\theta + \omega_y\,s_\theta \\ \omega_x\,\omega_y\,\nu_\theta + \omega_z\,s_\theta & \omega_y^2\,\nu_\theta + c_\theta & \omega_y\,\omega_z\,\nu_\theta - \omega_x\,s_\theta \\ \omega_x\,\omega_z\,\nu_\theta - \omega_y\,s_\theta & \omega_y\,\omega_z\,\nu_\theta + \omega_x\,s_\theta & \omega_z^2\,\nu_\theta + c_\theta \end{bmatrix},$$

1 0

где $c_\theta = \cos\theta$, $s_\theta = \sin\theta$, $\nu_\theta = 1 - \cos\theta$.

Для проверки на особые точки используйте следующие матрицы:

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad R = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \qquad R = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3.1. $R = \begin{bmatrix} -0.5092 & -0.0269 & 0.8602 \\ 0.7973 & 0.3617 & 0.4833 \\ -0.3242 & 0.9319 & -0.1628 \end{bmatrix}$

3.2. $R = \begin{bmatrix} -0.4434 & -0.4486 & 0.7760 \\ 0.8961 & -0.2012 & 0.3957 \\ -0.0214 & 0.8708 & 0.4912 \end{bmatrix}$

3.3. $R = \begin{bmatrix} -0.1350 & -0.1968 & 0.9711 \\ 0.7667 & -0.6416 & -0.0234 \\ 0.6277 & 0.7413 & 0.2375 \end{bmatrix}$

3.4. $R = \begin{bmatrix} 0.2233 & 0.7083 & 0.6697 \\ 0.7821 & -0.5402 & 0.3107 \\ 0.5818 & 0.4544 & -0.6746 \end{bmatrix}$

3.5. $R = \begin{bmatrix} 0.2117 & -0.0352 & 0.9767 \\ 0.6449 & -0.7459 & -0.1666 \\ 0.7344 & 0.6652 & -0.1352 \end{bmatrix}$

3.6. $R = \begin{bmatrix} -0.4777 & -0.3495 & 0.8060 \\ 0.8728 & -0.2932 & 0.3901 \\ 0.1000 & 0.8899 & 0.4451 \end{bmatrix}$

3.7. $R = \begin{bmatrix} 0.0028 & -0.0401 & 0.9992 \\ 0.8698 & -0.4929 & -0.0222 \\ 0.4934 & 0.8692 & 0.0335 \end{bmatrix}$

3.8. $R = \begin{bmatrix} -0.8299 & 0.2217 & 0.5120 \\ 0.3220 & -0.5592 & 0.7640 \\ 0.4557 & 0.7989 & 0.3927 \end{bmatrix}$

3.9. $R = \begin{bmatrix} -0.1631 & 0.9346 & 0.3162 \\ 0.9840 & 0.1773 & -0.0167 \\ -0.0716 & 0.3084 & -0.9485 \end{bmatrix}$

3.10. $R = \begin{bmatrix} 0.2696 & 0.2931 & 0.9173 \\ 0.9347 & 0.1496 & -0.3225 \\ -0.2317 & 0.9443 & -0.2336 \end{bmatrix}$

3.11. $R = \begin{bmatrix} -0.3565 & 0.0457 & 0.9332 \\ 0.9110 & 0.2386 & 0.3363 \\ -0.2073 & 0.9701 & -0.1267 \end{bmatrix}$

3.12. $R = \begin{bmatrix} -0.5114 & 0.6943 & 0.5064 \\ 0.8520 & 0.4865 & 0.1934 \\ -0.1121 & 0.5303 & -0.8404 \end{bmatrix}$

3.13. $R = \begin{bmatrix} 0.3491 & 0.1182 & 0.9296 \\ 0.8084 & 0.4637 & -0.3626 \\ -0.4739 & 0.8781 & 0.0663 \end{bmatrix}$

4

Напишите функцию quart_slert, чтобы продемонстрировать SLERT (spherical linear interpolation) двух кватернионов и промежуточных между ними.

- кватернион q_0 — начальная ориентация;
- кватернион q_1 — конечная ориентация;
- переменная steps — общее число кватернионов, включая промежуточные между q_0 и q_1.

Функция должна возвращать матрицу q_int размерности steps × 4, которая хранит все промежуточные кватернионы, включая сами q_0 (первая строка матрицы) и q_1 (последняя строка матрицы).

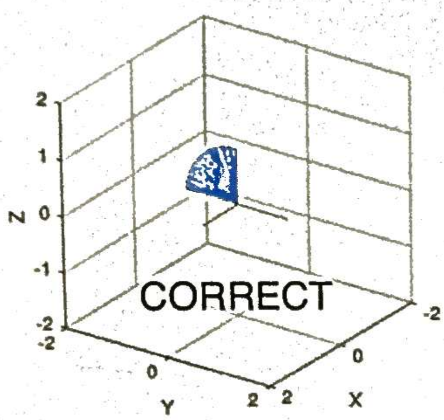
Угол между (единичными) кватернионами можно найти, используя скалярное произведение:

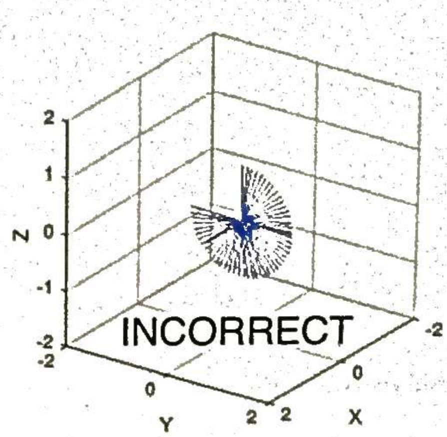
$$\cos\Omega = q_0 q_1.$$

Промежуточный кватернион в момент времени t вычисляется согласно формуле:

$$q_t = \frac{\sin((1-t)\Omega)}{\sin\Omega} q_0 + \frac{\sin(t\Omega)}{\sin\Omega} q_1.$$

Важно! При написании функции необходимо «найти» кратчайший путь между двумя поворотами:





Для тестирования алгоритма необходимо сохранить изменения в функции quat_slerp и запустить perform_slerp. Там же можно регулировать начальные значения q_0, q_1 и steps.